MCMASTER – MOHAWK JOINT VENTURE

BACHELOR OF TECHNOLOGY PARTNERSHIP

FOUR-YEAR UNIVERSITY DEGREE PROGRAMS

EXPERIMENT TITLE: 3 Level Project & 7 segment display

**SUBMITTED BY:**

Marcus J Cameron – 400319790

Rodney Mbaguta – 400233571

LAB SECTION NUMBER: L03

INSTRUCTOR: Rameez Ashraf

LAB STARTED DATE: November 12, 2024

LAB DEMO DATE: November 27, 2024

LAB SUBMISSION DATE: December 1, 2024

**PROJECT OBJECTIVE**

The objective of this lab project is to design and implement a 3-level elevator system utilizing digital logic circuits to control a DC motor and display the elevator's current level on a 7-segment display. The project involves using push buttons for user input, reed switches for feedback on the elevator's position, and a motor driver to control the movement of the motor. By integrating these components, students will gain hands-on experience in designing digital control systems, working with sensors and displays, and applying motor driver circuits in a practical, real-world application.

**PROJECT RESULT OVERVIEW**

A machine with wires on a table

Description automatically generated A machine with wires and wires

Description automatically generated

**Hardware and Trigger Table**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Trigger** | **Description** |
| Push Buttons (3) | When a PB is pressed. | - The cabin moves to the floor corresponding to the pressed push button. Each of the three push buttons corresponds to a specific floor (Floor 1, Floor 2, and Floor 3). Pressing a button triggers the movement of the elevator to the desired floor. |
| LEDs (3) | When the cabin reaches the floor. While the cabin moves. | - A solid LED light indicates the cabin has reached the floor.  - The LED blinks while the cabin is in motion, providing visual feedback to indicate that the elevator is moving between floors. This feature helps to inform users of the cabin’s current status. |
| 7 Segment Display (1) | When the cabin reaches the floor. | - The 7-segment display shows the current floor number of the cabin. This display updates each time the cabin reaches a new floor, giving users clear visual feedback on the floor the cabin has arrived at. |

**Additional Features Tables**

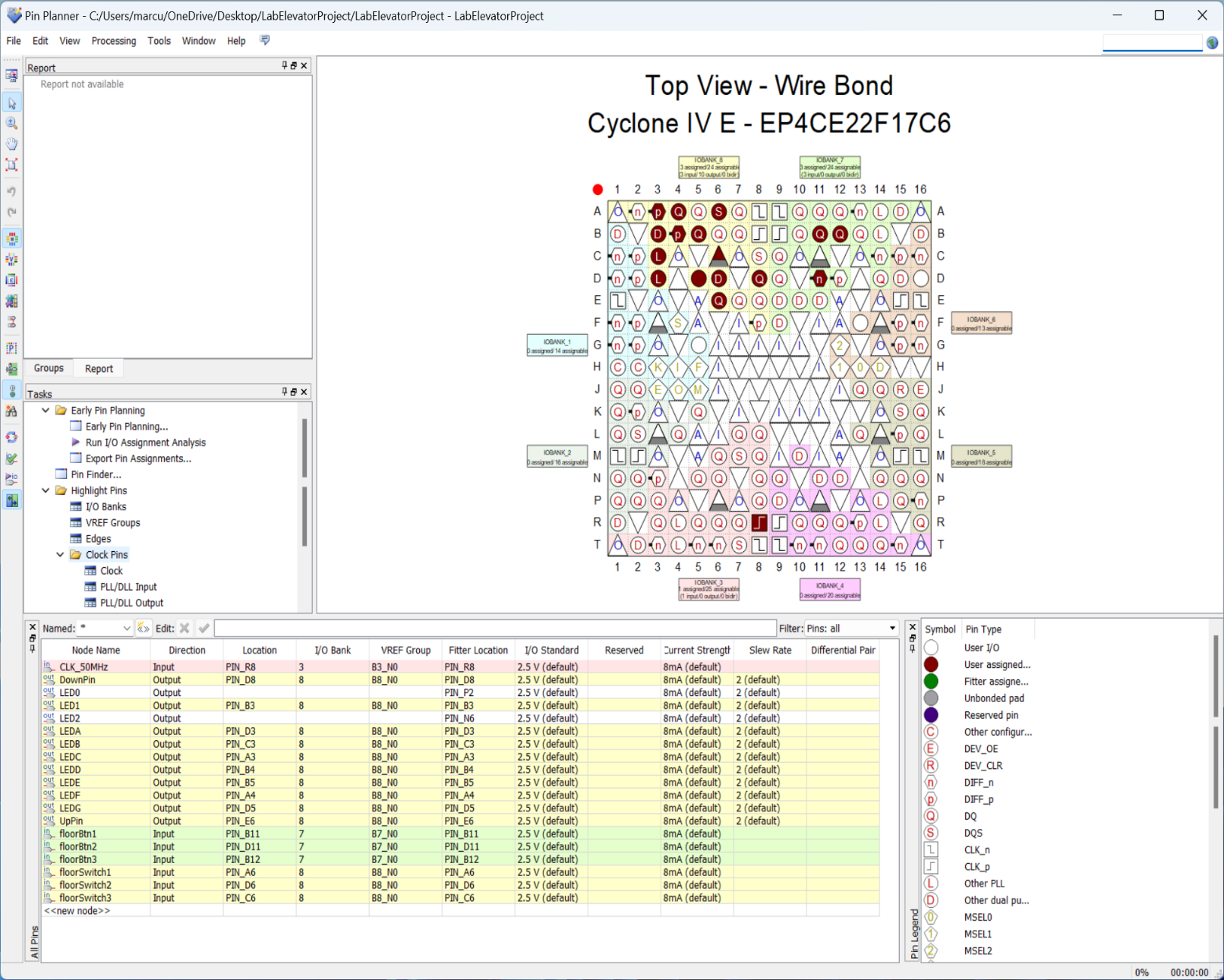
|  |  |  |  |
| --- | --- | --- | --- |
| Hardware | Trigger | Additional Feature | Description |
| 2 LEDs | Pushbutton is pressed, and the cabin moves either up or down. | 2 LEDs indicating the direction of the cabin. | - One LED lights up when the cabin is moving Up, and the other lights up when moving Down. |
| 7 Segment Display | Pushbutton is pressed to move the cabin to a desired floor. | Flashing Display | - The 7-segment display flashes the number of the floor the cabin is moving to. |
| Pushbutton, Reed Switches | Pushbutton is pressed, and the cabin automatically calibrates before going to the desired floor. | Calibration | - Cabin calibrates to the lower floor.  - 7-segment display shows '8' while flashing until it reaches the lower floor. |
| 3 Pushbuttons | When you press a push button while the cabin is already in motion. | Motion Lock | - Pressing a PB while the cabin is in motion does not redirect the cabin.  - The new input is accepted only once the cabin reaches the desired floor and is not in motion. |
| Reed switches, 3 more PB | When a desired floor’s reed switch does not detect the cabin. | Troubleshooting mechanism for reed switches. | - If the reed switch for a floor (e.g., floor 2) is not working or connected, the cabin will move to the next floor (e.g., floor 3) and recalibrate to try and find floor 2.  - Additional pushbuttons are used for troubleshooting, connected to the reed switches on floors 1, 2, and 3. |

**ELEVATOR ELECTRICAL SCHEMATICS**

A computer diagram with many green and red lines

Description automatically generated with medium confidence

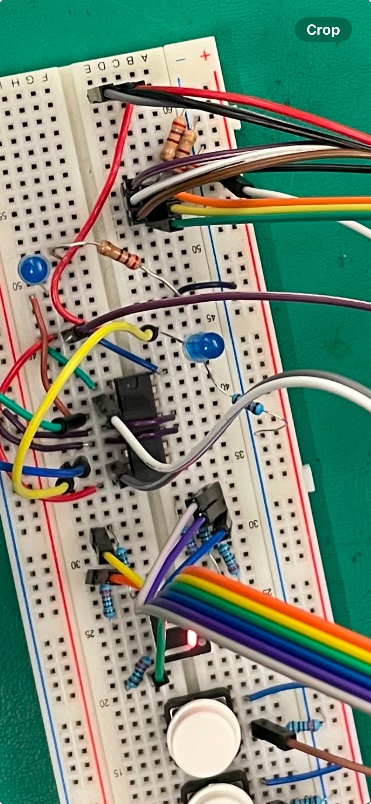
**FPGA PIN ASSIGNMENTS**



We used the following GPIO Pin:

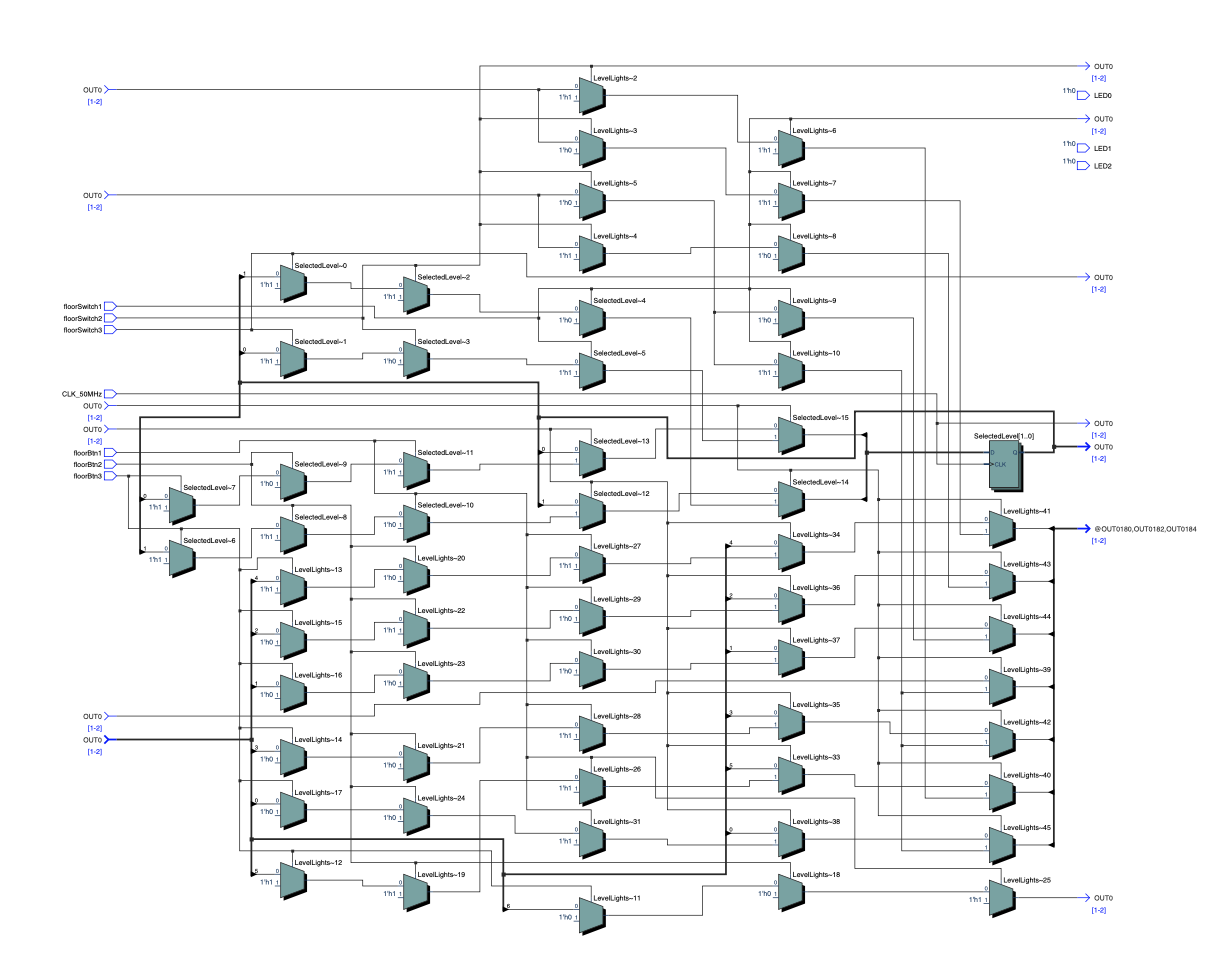
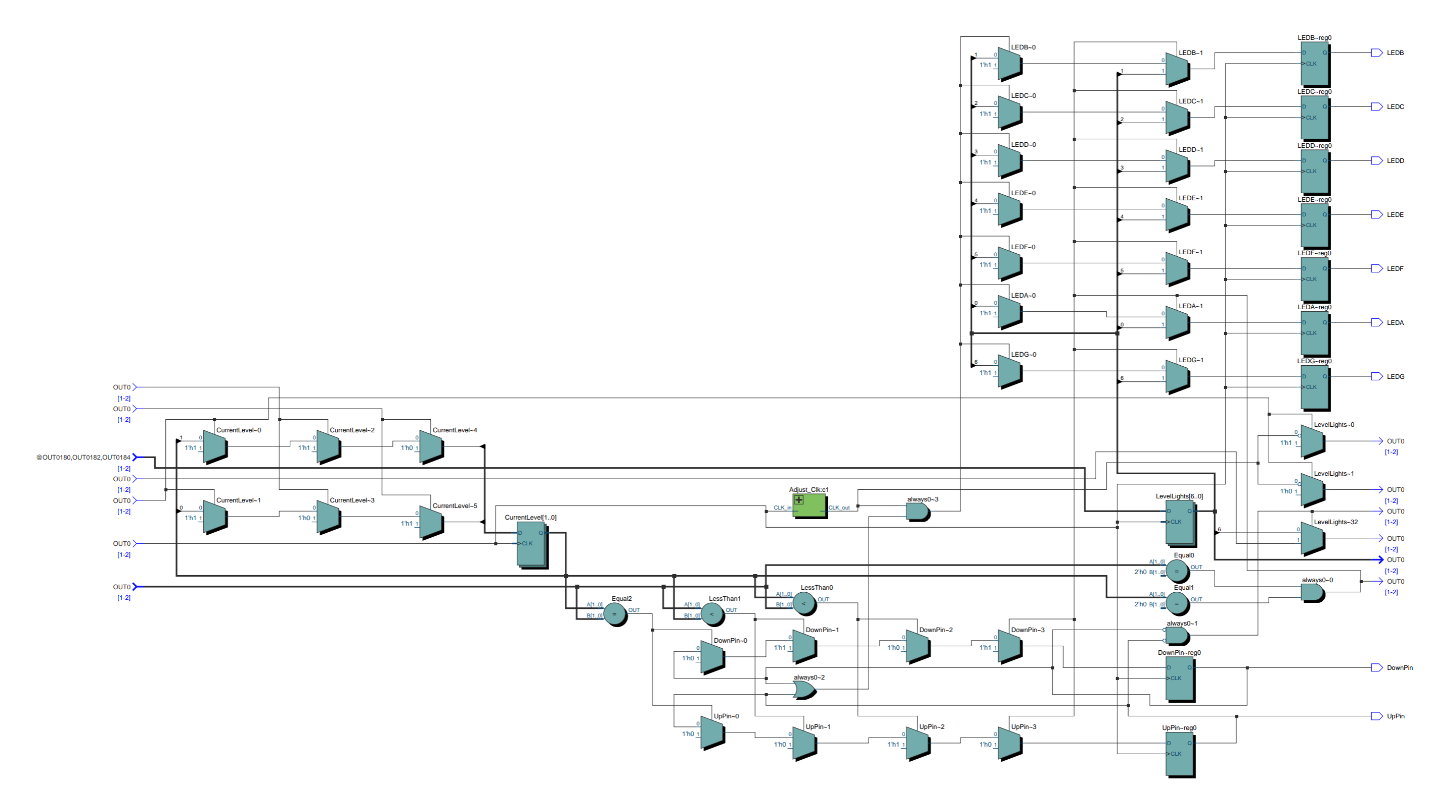
* Floor buttons 1, 2, and 3 are assigned to ports GPIO\_029, GPIO\_031, and GPIO\_033, respectively.
* Floor switches/reed switches are assigned to ports GPIO\_011, GPIO\_013, and GPIO\_015, respectively.
* Forward and reverse motor control pins are assigned to ports GPIO\_017 and GPIO\_019, respectively.
* 7-segment display LEDs are connected to ports GPIO\_00, GPIO\_01, GPIO\_03, GPIO\_05, GPIO\_07, GPIO\_06, and GPIO\_09, corresponding to segments A through G.

**INTEGRATION OF TEXAS INSTRUMENTS SN754410 QUADRUPLE HALF-H DRIVER**

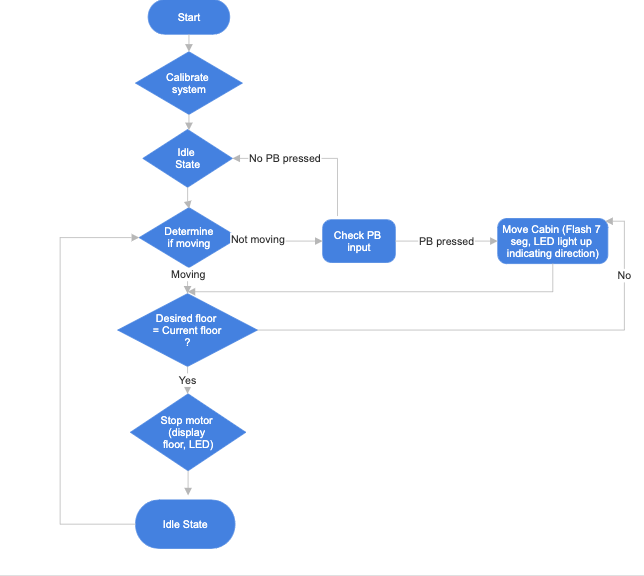


* Pin 1, 8, and 16 were connected to 5V+, supplied from the DE0-Nano FPGA’s VCC\_SYS GPIO Pin. This provides the necessary power for the logic circuits of the SN754410, ensuring proper operation of the driver in controlling the motor.
* Pins 2 and 7 control the forward and reverse motion of the elevator by receiving logic signals from the FPGA. By inverting the voltage at these pins, the outputs to the motor are also inverted, which results in reversing the polarity across the motor, making it move in the opposite direction. The voltage drop across the motor increases with higher voltage input to these pins, causing the motor to run faster. By adjusting the voltage applied to these pins, the speed and direction of the elevator can be precisely controlled.
* Pins 3 and 6 are used to power the motor. These pins provide the necessary current to drive the motor in the desired direction. The voltage supplied to these pins (via VCC2) determines the motor's operation, and the current drawn by the motor is controlled by the state of the enable pins (1,2EN and 3,4EN).
* Pins 4, 5, 12, and 13 were connected directly to the common ground of the FPGA. This common ground connection ensures that all components in the system share the same reference point for their electrical signals, preventing signal inconsistencies and ensuring stable operation of both the FPGA and the SN754410 driver.

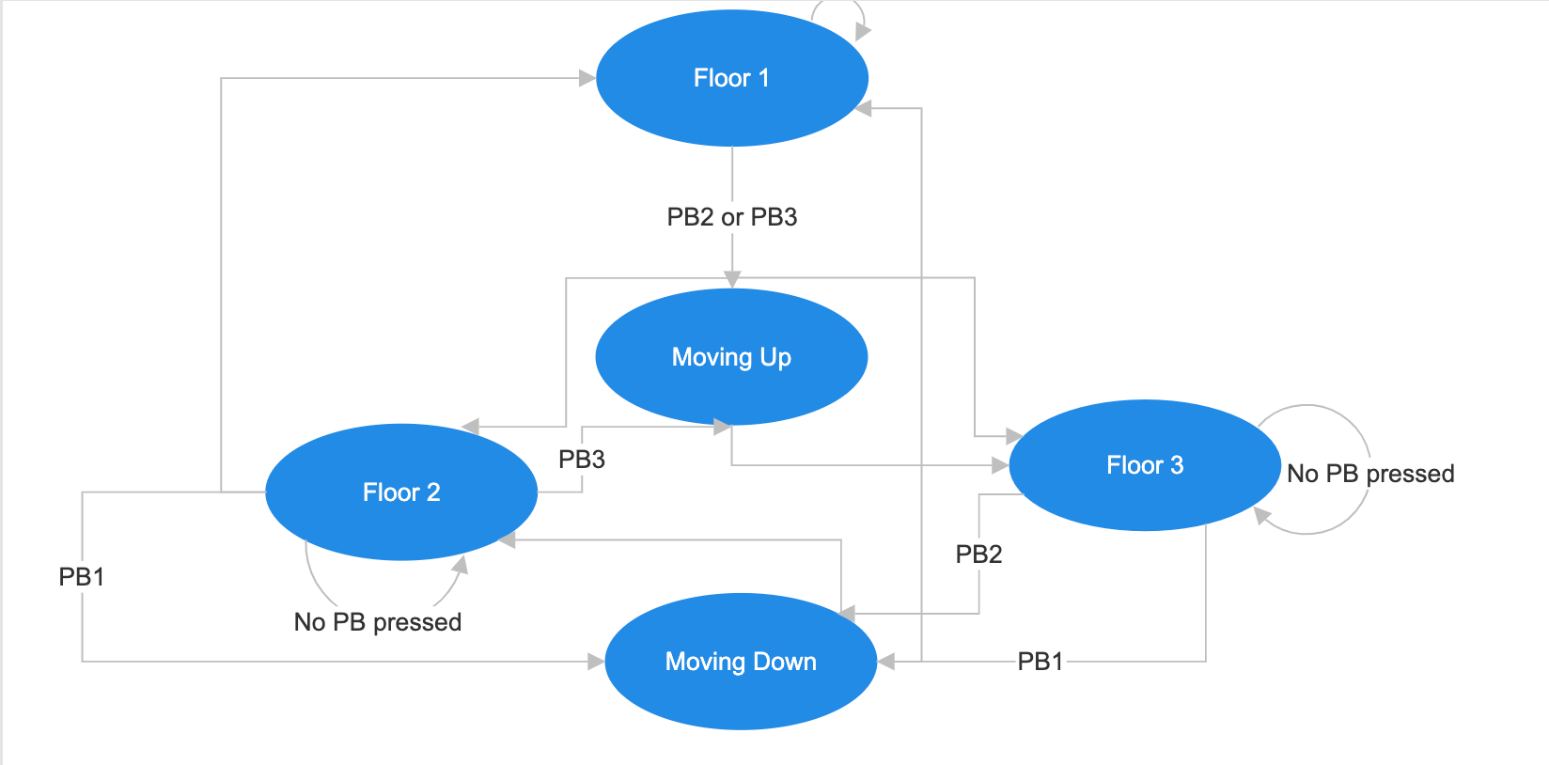
**RTL VIEW**



**FLOW CHART WITH DECISION BLOCK**



**STATE DIAGRAM**



**PROJECT LIMITATSION**

-Hardware dependency: we experienced some difficulties when testing the reed switches as some of them were not accurate and this would impact on our results and performance.

**PROJECT ACHEVIMENTS**

Throughout the project, we successfully implemented both the basic and advanced features of the elevator controller. The core functionality was achieved by designing the system to move the elevator to the desired floor when a **push button** was pressed, with the current floor number displayed on a **7-segment display**. This allowed users to easily identify which floor the elevator was on at any given time.

We also incorporated several advanced features, including:

* **Flashing 7-Segment Display**: The 7-segment display flashes when the elevator is in motion, providing a dynamic visual cue to indicate that the elevator is traveling between floors.
* **Calibration Feature**: A key feature of the system is the **calibration process**, which ensures that the elevator automatically returns to the **lower floor** when it is between two floors. This prevents the elevator from starting from an undefined position, ensuring accuracy in its movement.
* **Direction Indicators (LEDs)**: Two **LEDs** were added to indicate the direction of the elevator. One LED lights up when the elevator is moving **up**, and the other lights up when the elevator is moving **down**, providing users with real-time feedback about the elevator’s direction.
* **Motion Lock**: To ensure safe operation, we implemented a **motion lock feature**. This prevents the user from pressing a push button to redirect the elevator while it is already in motion. Any new input is only processed once the elevator

its destination and is stationary, preventing erratic behavior or malfunction.

This project allowed us to apply key concepts in both **combinational** and **sequential logic**. We designed and tested the system using the skills learned throughout the semester in our lab sessions, including working with digital logic components, Verilog programming, and FPGA interfacing. The experience not only deepened our understanding of hardware design but also provided valuable hands-on practice with troubleshooting and optimizing a complex digital system.